Common Sense Validation

Sumeha Kashyap [1,*], Sangeetha Balasubramanian[2,*], Anupam Yadav[3,*], and Aarshee Mishra[4,*]

[*]*College of Information and Computer Science, University of Massachusetts, Amherst*

[1]sumehakashya@umass.edu

[2]sangeethabal@umass.edu

[3]anupamyadav@umass.edu

[4]aarsheemishr@umass.edu

1 Problem statement

The aim of general artificial intelligence is to build machines with more human like capabilities. While there are domains in which machines have achieved super-human skills, such as object recognition, there are tasks in which they are far-off. One such area is common sense reasoning, where we outperform the machines by a large margin. Research on common sense understanding systems has received increased attention. The aim of research in this domain is to have more natural interactions between humans and machines. Progress in this field has far reaching consequences, from virtual voice assistant to understanding what is in the picture.

Common sense reasoning is a difficult task for a machine. Consider the statement 'getting a PhD is easy'. For humans it is easy to refute this. But for machines, to answer this they would need to know:

- What a PhD is?
- How much time it typically takes to get a PhD?
- What is the average income of a person doing a PhD?
- What is the lifestyle of a typical PhD student?
- What is the drop-out rate in a typical PhD program?

As in the example above, natural language sentences can often be syntactically and grammatically correct but may not make any sense. Common sense reasoning is one of the main bottle necks in machine intelligence. There are quite a few benchmark tasks and datasets such as (Rashkin et al., 2018a) and (Zellers et al., 2018) that evaluate a system's ability to infer commonsense knowledge in order to reason and understand natural language text. In our work we analyze some of the existing approaches and benchmarks to differentiate natural language statements that make sense from those that do not make sense. Given two statements, our work introduces a novel way of differentiating them on the basis of commonsense.

2 What you proposed vs. what you accomplished

Following are the tasks we set out to accomplish. Crossed out items refers to the tasks we accomplished:

- ~~Implement a Naive Bayes Classifier baseline~~
- Verification of the claim of achieving 70.1 accuracy using pretrained BERT Language Model made by (Wang et al., 2019): According to the paper, "*we calculate perplexities of both statements, choosing the one with lower scores as the correct one*". Since we cannot compute perplexity of sentence from BERT in a straightforward way, we did not attempt this approach.
- ~~Train a binary neural network classifier on BERT embeddings for single sentences.~~
- ~~Train a binary neural network classifier on BERT embeddings for pair of sentences.~~
- ~~Incorporate external knowldege from various knowledge bases like ConceptNet to word embeddings.~~
- ~~Analyze and compare the performances of the two classifiers with and without knowledge graphs.~~

3 Related work

Common Sense Reasoning is crucial to Natural Language understanding. Problems like co-reference resolution, reading comprehension, event prediction require systems to implicitly have some level of common sense to perform well. Several data sets have been developed that indirectly test a systems common sense reasoning ability. One such data set is the Wino-grad Schema Challenge (WSC) (Levesque et al., 2012) which requires answering a multiple choice test. Each example consists of 3 parts:

- A brief discussion of a topic, consisting of an ambiguous pronoun.
- A question about the ambiguous pronoun.
- Answer choices regrading the question.

Current state of the art approach (Prakash et al., 2019), combines existing language models such as BERT (Devlin et al., 2018), with a commonsense knowledge hunting module.

Another dataset is Choice of Plausible Alternatives (COPA)(Roemmele et al., 2011) where given a premise and two answers, the task is to choose the more plausible of the two. Current state of the art (Li et al., 2019) for this dataset uses BERT with a margin based loss function.

In the JHU Ordinal Common-sense Inference (JOCI) (Zhang et al., 2017), given context and hypothesis, task is to score the likelihood of the hypothesis in 5 levels from very-likely to technically impossible. In Event2Mind (Rashkin et al., 2018a), model is tested based in probable reactions of the event participants. Stating it formally, given an event described the task is to generate possible intents and the reactions of the participants. State of the art (Du et al., 2019) proposes, Context-aware Variational Autoencoder trying to learn the background information. Situations with Adversarial Generations (SWAG) (Zellers et al., 2018) has multiple choice questions over a wide variety of scenarios, with BERT (Devlin et al., 2018) as current state of the art approach.

Each of the dataset above mentioned have their own nuances and it is not obvious which dataset is best suited to evaluate the task of common sense evaluation. Though, the Common Sense Validation task addressed in this proposal scores common sense more directly and in the second task asks for the plausible explanations.

Existing state of the art language models such as BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), GPT-2 (Lagler et al., 2013) and ELMo (Peters et al., 2018) being trained on large datasets should already have some notion of common sense. Moreover as seen in the above mentioned datasets, most of the state of the art models for the task use one of these models.

For futher improving performance on common sense reasoning tasks , large-scale common-sense inference knowledge resources have been used in the recent years. Atomic (Sap et al., 2019) presents a huge everyday commonsense reasoning knowledge graph, which has nine if-then relations with variables, including causes, effects, and so on. Event2Mind (Rashkin et al., 2018b) proposes a new corpus and task, aiming to find out mentioned/unmentioned people's intents and reactions under various daily circumstances. These datasets are not directly useful for our benchmark since they focus only on a small domain. ConceptNet is a prestigious knowledge graph that has been upgraded over a long time ((Liu and Singh, 2004); (Havasi et al., 2007); (Speer et al., 2017)). ConceptNet constructs triples using labeled edges as relations and various words and/or phrases as entities. It also has the sentences describing the corresponding triples.

(Wang et al., 2019) introduced the Common Sense Validation task and reported the performance of few baseline models. For the sense making task, they calculate per-plexities of both statements, choosing the one with lower scores as the correct one. For explanation, they concatenate the statement with the each reason and then use the three concatenated sentences to calculate perplexities. They evaluated the performance of this setup with pre-trained BERT features , pre-trained ElMo features and fine-tuned ELMo features. Bert performs better on the explanation task while ELMo preforms better on the classification task.

4 Your dataset

Task. Given a pair of natural language statements we identify which of them makes more sense. Both sentences have similar syntactic structures differing only by a few words. These pairs of sentences are used on our first subtask called Sen-Making, which requires the model to identify which one is valid. Some examples of sentences that don't make sense are: *'He poured or-*

ange juice on his cereal', 'He drinks apple', 'Jeff ran 100,000 miles today'. While corresponding sentences that make sense are: *'He poured milk on his cereal', 'He drinks milk', 'Jeff ran a mile today'*.

Data. In our dataset, we have a total of 12021 pairs of sentences. The entire data has been annotated and is available online[1]. The dataset (Wang et al., 2019) has been released as part of the SemEval 2020 contest. We refer to the first sentence as *sent0* and second one as *sent1*.

The total number of words in our vocabulary is 9018. The average length of the sentences is 39.3±13.6. The distribution of sentence lengths of the two classes is shown in figure 1. The average sentence length difference between the two classes is 2.8±3.1. Figure 2 shows the absolute difference in word count between two classes in decreasing order. We notice that incorrect statements have almost the same negative different words compared with correct statements. The incorrect statements have 53 'don't' or 'can't' or 'not' or 'no' as different words, while the correct statements have 55. This can illustrate that the corpus does not use negative words to construct incorrect statements or correct statements.

In our test set, we have 2021 sentence pairs. In 1352 pairs both the sentences have the same length and differ by just one word, occurring in the same place in both sentences.

Data Preprocessing. For our baseline approach, Naive Bayes, we maintain a vocabulary of all the words in our dataset. For each word in our dataset we count its occurrences and store it in a map.

BERT has a constraint on the maximum length of a sequence after tokenizing. For any BERT model, the maximum sequence length after tokenization is 512. But we can set any sequence length equal to or below this value. For faster training, we be using 128 as the maximum sequence length and also since our sequences are not too long. An input feature consists of purely numerical data (with the proper sequence lengths) that can then be fed into the BERT model. This is prepared by tokenizing the text of each example and truncating the longer sequence while padding the shorter sequences to the given maximum sequence length (128).

5 Baselines

(Wang et al., 2019) choose state-of-the-art language models trained over large texts as the baselines, assuming that common sense knowledge are encoded over texts. For the sense making task, they calculate perplexities of both statements, choosing the one with lower scores as the correct one. For explanation, they first concatenate the statement with the each reason and then use the three concatenated sentences to calculate perplexities. For example, we concatenate "he put an elephant into the fridge" with its optional reasons to be "he put an elephant into the fridge" is against common sense because an elephant cannot be put in a fridge.

As shown in Table 1, ELMo and BERT have a significant advantage over random results in Sen-Making. (Wang et al., 2019) conjure that both of them have the ability to judge whether a sentence is with or against common sense. For the task of Sen-Making, ELMo does better than BERT. Fine-tuned ELMo has an obvious improvement in Sen-Making, probably because introducing knowledge will help models to identify common sense but cannot help them in inference. However, fine-tuning makes BERT perform the same in Sen-Making. The authors hypothesize that this is likely because the original BERT models trained on BookCorpus (Zhu et al., 2015) and English Wikipedia contain sufficient common knowledge and the fine-tune operation may be useless or even makes the models be specific in the finetuning corpora; Besides, fine-tuning may corrupt the structure formed by Next Sentence Prediction.

We would also like to point out that this dataset is being used in the *SemEval 2020 Task 4 - Commonsense Validation and Explanation* contest. The Sen-Making task is subtask A in this

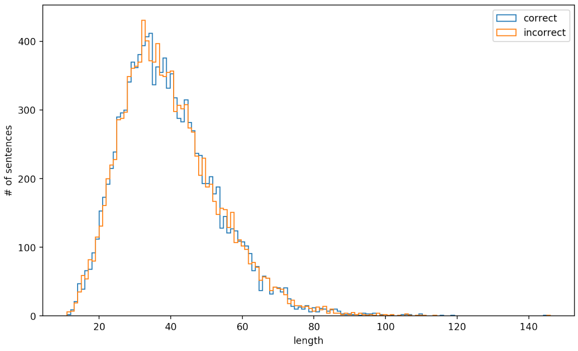

Figure 1: Distribution of sentence lengths.

[1] https://github.com/wangcunxiang/Sen-Making-and-Explanation

competition for which groups have accuracies as high as 100% [2]! The focus of (Wang et al., 2019) was to introduce the dataset and there results on the Sen-Making task are dwarfed by most entries in the contest as well as by our models (explained in the Approach section).

We have implemented one baseline model for the dataset which is a naive-bayes classifier. We pre-process the input text using white space tokenization to create a bag-of-unigrams representation. The dataset is split into train and test sets in the ratio of 83:17. This results in 10000 training examples and 2021 test examples. The training set is split further to include a validation set of 2000 examples (20% of the training data). Our evaluation metric is accuracy which is determined by the ratio of ‘number of sentence pairs in which both the sentences are correctly classified’ to ’the total number of sentence pairs’. The hyperparameter for our baseline model is the smoothing parameter α. Tuning the parameter α, we observe that the accuracy for the validation set increases sharply for small values of alpha and then dips gradually as alpha increases. For $\alpha = 2$ we observe the test accuracy to be 56.9%.

It is interesting to note also that unlike reading comprehension tasks (Rajpurkar et al., 2016), where machines can surpass human performance by careful finetuning, it remains a big challenge for systems to reach human performance, which is near 100% for sense making. When the human testee is asked to look at the mistakes that they make, we find that most human errors are due to reduced concentration rather than conceptual issues.

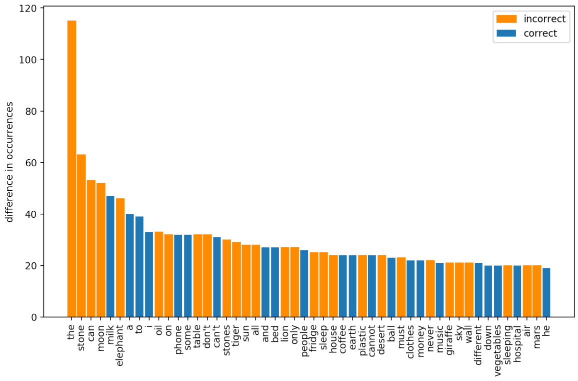

Figure 2: Absolute difference in word count.

6 Your approach

We have implemented and evaluated 3 approaches for our task. The best approach gives an accuracy of 93.47%.

6.1 BERT (Masked LM)

Our first approach is inspired by the fact that almost fifty percent of the examples in the dataset have the same length for *sent0* and *sent1* and differ by only one token. This approach is implemented only on the sentences which follow the above rule. For every sentence pair, we mask out the word that is different between the two sentences and predict the probability distribution of this MASK over the list of words in the vocabulary using a pre-trained BERT(Devlin et al., 2018) language model. Using this distribution, we use the most probable word as the actual word for the sentence. The sentence which has a higher probability for the actual word is classified as ‘makes sense’ while the other as ‘does not make sense’. We expected the pre-trained BERT model to perform exceedingly well for this setup because BERT was trained on a very similar task (Masked Language Model) in (Devlin et al., 2018).

Although evaluation in this setup does lead to a higher accuracy compared to the baseline implementation of (Wang et al., 2019), it is not on the scale of improvement we were expecting. The underwhelming results combined with the fact that this type of evaluation is applicable only to a subset of the data (sentence pairs which follow the rule described above), we decided to use the following approaches.

6.2 BERT (Sentence Pair)

In the second approach, we fine-tune BERT for Sequence Classification. For every training example, we use *sent0* as the first sentence input and *sent1* as the second sentence input. The targets for the classification task are binary labels. The label is ‘1’ when the *sent0* is the sentence that ‘makes sense’ and ‘0’ otherwise. We tuned the initial learning rate and the total number of epochs to make the model converge. On using the default BERT parameters (‘weight-decay’: 0, ‘learning-rate’: 4e-5,‘adam-epsilon’: 1e-8,‘train-batch-size’: 8, warm-up steps :0) and setting total epochs to 1, causes the loss to hover around the initial value after the epoch is over. However changing the total epochs to 10, made the loss fall

[2] https://competitions.codalab.org/competitions/21080#results

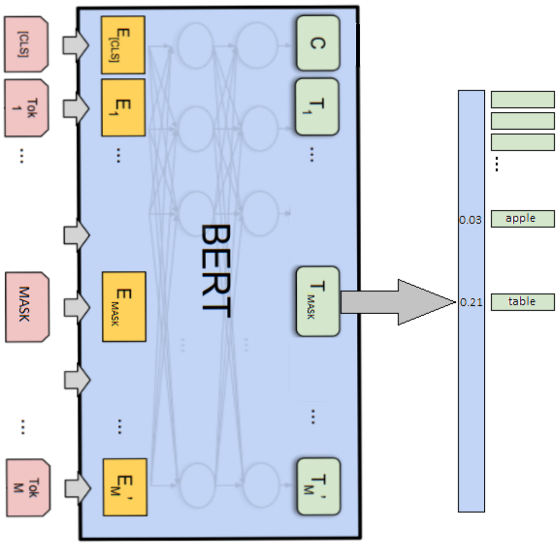

Figure 3: For Masked Language Modelling evaluated on a subset of the data

after 1 epoch. We realized that this happened because, traditionally BERT uses a Linear Scheduler, and the rate of fall of the learning rate depends on the total number of epochs. Thus setting the total epochs to a larger number, makes the fall of learning rate less extreme and helps in model convergence. We assume that the model converges when the training loss stops changing. This model gives an accuracy of 88.71%.

Note that this model leads to a significant improvement in performance over all the baseline implementations and also our first approach described above. This leads us to believe that BERT has the capability to judge whether a sentence is with or against common sense. This may be due to the large amounts of common sense knowledge present in the data/raw text (English Wikipedia, BookCorpus (Zhu et al., 2015)) the original BERT model was trained on.

6.3 BERT (Single sentence)

For our third approach, we again fine-tune BERT for Sequence Classification but change the input format of the model. We do not use the second input to BERT for Sequence Classification. For every example in the dataset, we used both *sent0* and *sent1* as the first input to BERT in an iterative fashion. The targets for the classification task are the binary labels similar to the previous approach. While inference, we compare the scores for the individual sentences that belonged to a sentence pair

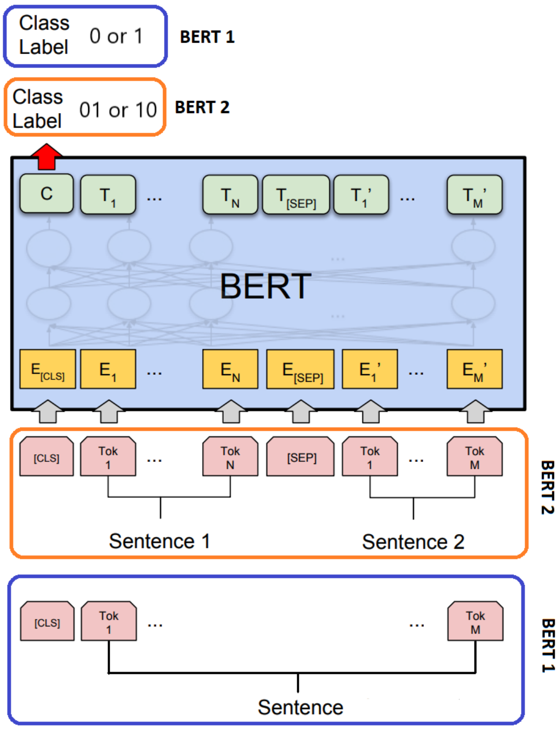

Figure 4: BERT For Sequence Classification fine-tuned using single/multiple sentence(s) input

in the data. The sentence with the higher score is assigned the label '1', i.e. 'makes sense' and the other sentence in the pair is given the complimentary label '0', i.e. "does not make sense'. At test time, the evaluation is done based on whether the model is able to correctly classify both individual sentences that belonged to a sentence pair. For example, we use sent0 *"We will get pins and needles from sitting or standing for long periods of time"* and sent1 *"We will feel relaxed from sitting or standing for long periods of time"* as the first input to BERT for Sequence Classification. We then observe the unnormalized score obtained for both these sentences. The sentence *"We will get pins and needles from sitting or standing for long periods of time"* receives a higher score and we assign this sentence the label '1', i.e. 'makes sense' and assign the label '0', i.e. 'doesn't make sense' to the sentence *"We will feel relaxed from sitting or standing for long periods of time"*. We shuffle the dataset so that sentences that belong to the same dataset example do not always occur one after the other. As in the previous approach, we tuned the initial learning rate and the total epochs to achieve convergence. This model gives an accuracy of 93.47%.

Clearly this model outperforms the naive BERT for Sequence Classification approach used in the previous method described above. This result is peculiar as BERT is able to classify sentences with a higher accuracy when they are treated as individual training examples to the classifier. We hypothesize this is the case because of the modified classification objective defined by us above. While the raw scores obtained are independent for each sentence, the way we assign the binary labels forces the model to compare sentences that belonged to a pair in the original test example. This helps improve the performance of the model to distinguish between sentences that were both receiving high or low scores. Since we are choosing the sentence with the higher score, we are able to correctly classify sentence pairs where the model was interpreting both sentences to have the same 'sense'.

6.4 ERNIE

For incorporating knowledge base information, we used ERNIE: Enhanced Representation through Informative Entities (Zhang et al., 2019). ERNIE uses wikidata as the knowledge graph to incorporate knowledge information into BERT. To get the entity mentions from sentences , we used TAGME[3]. We discarded the entities selected using TAGME which had an entity mention confidence score lower than 0.3. Ernie masks the entities for which entity mention cannot be found and pads the input tokens in a manner similar to Bert by using an input mask. Each entity embedding was 100 dimensional. This was done in the sentence pair style method(described previously), by sending both sentences in each input example as sent_a and sent_b respectively and performing binary classification , where if sent_b makes more sense than sent_a , the label is 1 and vice versa.

We expected this model to outperform all the previous models. However, as seen from Table 1, ERNIE lags far behind in terms of performance when compared to both BERT models. This could be because we did not fine-tune the hyperparameters of the model enough due to compute restrictions on Colab.

One of the challenges we faced while doing this project is the dearth of compute. We did all of our training on Google Colab, because large size of the models we encountered issues like poor execution speed, freezing and crashing of the system.

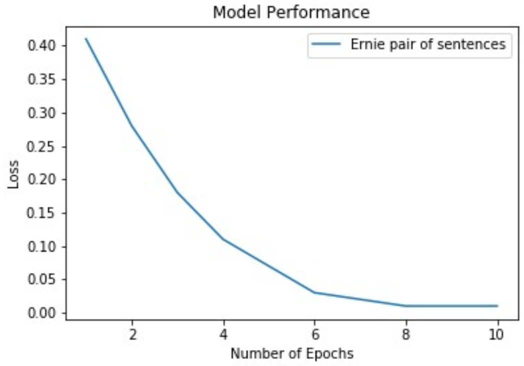

Figure 5: ERNIE cross-entropy training loss

Model	Accuracy (%)
Naive Bayes (Baseline)	56.9
BERT (Sentence Pair)	88.71
BERT (Single Sentence)	93.47
BERT (Wang et al., 2019)	70.1
BERT (Masked LM*)	77.27
ERNIE (Sentence Pair)	81.8

Table 1: Test results for the different models used for the classification task.*Masked LM is for sentence pairs that differ by a single word only (constitutes 50 percent data)

7 Error analysis

Table1 and Table 2 show the results of our approaches.

1. Bert Language Model -
   Examples of incorrectly classified sentence pairs of the same length that differ by one word-
   (a) She drove her children to the moon, She drove her children to the playground
   (b) He caught a cold and had a hot shower , He caught a cold and had a cold shower
   (c) he brushed his teeth at home , he brushed his teeth at school
   (d) I was playing basketball with a ball, I was playing basketball with a rock

   Both the sentences here are pretty obvious and a human being would have no difficulty in detecting the more logical statement. As explained before, we expected our masked LM model to shatter test cases like these because of BERT being originally trained explicitly on the masked language model objective. This goes to show that BERT doesn't

[3] https://tagme.d4science.org/tagme/

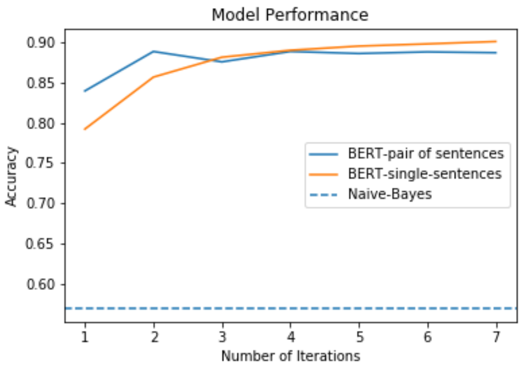

Figure 6: Validation Accuracy vs epochs

have a lot of knowledge about the entity relationships in a sentence. It fails to understand relationships like basketball is played with a ball and that people don't drive to the moon!

2. Bert for sentence pair classification -
   The model fails on 228 sentence pairs out of 2021 sentence pairs in our test set. Out of these 228 incorrectly classified sentence pairs, we notice that there are 155 sentence pairs that have same length and differ by only one word at the same position. Examples of incorrectly classified sentence pairs of the same length that differ by one word-

   (a) The car is driving slowly on the highway , The car is driving fast on the highway
   (b) He caught a cold and had a hot shower , He caught a cold and had a cold shower
   (c) he brushed his teeth at home , he brushed his teeth at school
   (d) A boy is a male , A girl is a male
   (e) The south pole is very hot , The south pole is very cold

   Examples of incorrectly classified sentence pairs that differ in more than one word -

   (a) Animals are pets, Pets are animals
   (b) People usually fly in the sky , People usually walk in the land
   (c) The tortoise usually runs faster than the rabbit , The rabbit usually runs faster than the tortoise
   (d) Vegetarian usually like eating meat, Vegetarian usually don't eat meat

   Some of the misclassified sentence pairs are actually ambiguous and require more context (The car is driving slowly on the highway, The car is driving fast on the highway). In this case both sentence are not only gramatically and syntactically correct but also make sense! It is plausible that a car is driving slow on the highway and the sentence independently makes sense. The model needs external knowledge to understand that cars usually drive fast on the highway. This example highlights another subtlety that the model must possess: if both sentences in a pair make sense in the real world independently, then the model must also be able distinguish as to which sentence makes more sense. Capturing this capacity in the model was our main motivation for the BERT (Single Sentence Classification) which treats the classification of each sentence in a pair independently and then tries to figure out which sentence makes more sense. There are also some trivial sentence pairs (A boy is a male, A girl is a male) as well that are getting misclassified.

3. BERT for Single Sentence Classification -
   Examples of incorrectly classified sentence pairs that differ by only one word: This model fails on 131 sentence pairs in our test set. Out of these 131 pairs, 73 pairs have the same sentence length and differ by only one word. Since this model does not treat examples in a pairwise manner, we have calculated the accuracy such that a sentence pair is said to be correctly classified if both the sentences are classified correctly and incorrectly classified otherwise.

   (a) He walks in from the wall, He walks in from the door
   (b) People usually work on Sundays, People usually rest on Sundays
   (c) December is the 12th month of a year, December is the 13th month of a year
   (d) he opens the door with a lock he opens the door with a key

   Examples of incorrectly classified sentence pairs with different lengths:

   (a) I cooked my meal at the restaurant, I paid for my meal at the restaurant
   (b) A lemon tastes sour when it goes bad, Milk tastes sour when it goes bad

(c) I went underwater and held my breath, I went underwater and took a deep breath

(d) we should borrow things after returning them , we should return things after borrowing them

The BERT for Single Sentence Classification is the model that achieves the best accuracy on the test set. Analyzing the failure cases here shows that the model is primarily struggling with sentence pairs that have a dependence on knowledge that is derived from relationship between entities in the real world. For example, the sentence pair *"December is the 12th month of a year", "December is the 13th month of a year"* needs external context or knowledge that a year only has 12 months. The same is applicable to examples like *"he opens the door with a lock", "he opens the door with a key"* where information is needed about the functioning of a locking mechanism.

4. ERNIE for Sentence Pair Classification -
The model fails on 368 sentence pairs out of 2021 sentence pairs in our test set. Out of these 368 incorrectly classified sentence pairs, we notice that there are 185 sentence pairs that have same length and differ by only one word at the same position. Examples of incorrectly classified sentence pairs of the same length that differ by one word-

(a) A salad usually contains grass, a salad usually contains lettuce

(b) most children hate candies, most children love candies

(c) I'm hungry for water, I'm hungry for food

Examples of incorrectly classified sentence pairs with different lengths:

(a) most people become wiser after drinking a lot alcohol, most people become more stupid after drinking a lot alcohol

(b) We will get pins and needles from sitting or standing for long periods of time , We will feel relaxed from sitting or standing for long periods of time

ERNIE incorporates knowledge graph information into the BERT representations and we expected this model to outperform all the previous models. However, as seen from Table 1, ERNIE lags far behind in terms of performance when compared to both BERT models. This seems to be substantiated by the failure cases as well. Integrating Knowledge Graphs should've enabled the model to understand entity relationships in the sentences and this information should've helped it classify sentence pairs like *"A salad usually contains grass, a salad usually contains lettuce"*. The model is still not able to classify these types of sentences and, in fact, struggles to classify simpler sentence pairs like *"most children hate candies, most children love candies"* that were classified correctly by the BERT for Sentence Pair Classification model.

8 Contributions of group members

We all contributed equally to the project. Throughout the project we randomly assigned sections of the report/poster for writing. Major tasks performed by each member:

- Sumeha Kashyap:
  - Trained ERNIE (Zhang et al., 2019)
  - Research on prior work
- Sangeetha Balasubramanian :
  - Trained BERT sentence pair
  - Error analysis of the output
- Anupam Yadav:
  - Trained BERT single sentence
  - Data Analysis and Processing
- Aarshee Mishra:
  - Trained Naive Bayes
  - BERT Language Model inference

9 Conclusion

We started with Naive Bayes as our baseline model, and observed the accuracy of 56.9% on the test set. Since any random model can give us an accuracy of 50% this was giving us just above random performance.
(Wang et al., 2019) reported a similar performance for BERT before and after fine-tuning the model

for the sense-making task, and decrease in performance for explanation task. This lead us to believe, that BERT language model, without training should work pretty well on the pair of sentences of same length and differing by one word(similar sentences). For these sentences we were expecting pretty similar accuracy for both fine-tuned and untrained language model. Therefore, initially we thought of having an ensemble of models with untrained BERT Language model, for similar sentences and another model for rest of the sentences. But fine-tuned BERT Language model gave much better performance for compared to untrained model. The reason behind such a discrepancy could be the difference in prediction model. While they are trying to estimate perplexity of a sentence, we trained a fully connect ted model on top of `CLS` token.

Apart from this one surprising result we obtained was that BERT language model trained on the pair of sentences performed worse compared to the language model trained on only one sentence. This could be due to the fact that since we have two times the more data(after splitting) our model shoes better convergence.

We also expected the language model with wiki knowledge base, ERNIE (Zhang et al., 2019) to have better performance than the simple language model. This could be because we did not fine tune the hyperparameters of the model enough. When we tried to replicate the results from (Wang et al., 2019), we realized that using BERT it is not possible to obtain the perplexity score of the sentence, since even if we mask the entire sentence , it would have information about the length of the sequence, through the number of masked tokens.

Given more time, we would like to try incorporate other knowledge bases such as ConceptNet (Speer et al., 2017) to improve the performance of our system. Also since the two given pair of sentences are very closely related to each other we can also explore the use of biattentive classification networks (McCann et al., 2017) for our task. Also we have only looked into one task for the challenge i.e. Sense-Making, we can also look into other task of Explanation.

References

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Du, L., Ding, X., Liu, T., and Li, Z. (2019). Modeling event background for if-then commonsense reasoning using context-aware variational autoencoder. *arXiv preprint arXiv:1909.08824*.

Havasi, C., Speer, R., and Alonso, J. (2007). Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. In *Recent advances in natural language processing*, pages 27–29. Citeseer.

Lagler, K., Schindelegger, M., Böhm, J., Krásná, H., and Nilsson, T. (2013). Gpt2: Empirical slant delay model for radio space geodetic techniques. *Geophysical research letters*, 40(6):1069–1073.

Levesque, H., Davis, E., and Morgenstern, L. (2012). The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*.

Li, Z., Chen, T., and Van Durme, B. (2019). Learning to rank for plausible plausibility. *arXiv preprint arXiv:1906.02079*.

Liu, H. and Singh, P. (2004). Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

McCann, B., Bradbury, J., Xiong, C., and Socher, R. (2017). Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6294–6305.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL*.

Prakash, A., Sharma, A., Mitra, A., and Baral, C. (2019). Combining knowledge hunting and neural language models to solve the winograd schema challenge. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6110–6119.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Rashkin, H., Sap, M., Allaway, E., Smith, N. A., and Choi, Y. (2018a). Event2mind: Commonsense inference on events, intents, and reactions. *arXiv preprint arXiv:1805.06939*.

Rashkin, H., Sap, M., Allaway, E., Smith, N. A., and Choi, Y. (2018b). Event2Mind: Commonsense inference on events, intents, and reactions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 463–473, Melbourne, Australia. Association for Computational Linguistics.

Roemmele, M., Bejan, C. A., and Gordon, A. S. (2011). Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Sap, M., Le Bras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Roof, B., Smith, N. A., and Choi, Y. (2019). Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.

Speer, R., Chin, J., and Havasi, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Wang, C., Liang, S., Zhang, Y., Li, X., and Gao, T. (2019). Does it make sense? and why? a pilot study for sense making and explanation. *arXiv preprint arXiv:1906.00363*.

Zellers, R., Bisk, Y., Schwartz, R., and Choi, Y. (2018). Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.

Zhang, S., Rudinger, R., Duh, K., and Van Durme, B. (2017). Ordinal common-sense inference. *Transactions of the Association for Computational Linguistics*, 5:379–395.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. (2019). ERNIE: Enhanced language representation with informative entities. In *Proceedings of ACL 2019*.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.